The **asymptote** package

John Bowman, Tom Prince, and Will Robertson

2011/08/17   v1.25

Abstract

This package provides integration of inline and external Asymptote graphics within a LaTeX document.

Contents

1 Introduction

This is the documentation for the LaTeX package `asymptote` which accompanies the Asymptote drawing package. For further details on Asymptote, please see its documentation in `asymptote.pdf`.

2 User syntax

2.1 Package loading and options

The package may take two options at load time: `inline` or `attach`. These options can also be set at any time with the `\asysetup{⟨options⟩}` command, or specified individually in the optional argument to each `asy` environment or `asyinclude` command.

The `inline` option uses Asymptote's 'inline' mode whereby included graphics have their labels typeset in the environment of the document they are contained within. Otherwise the Asymptote graphics are self-contained and their formatting is independent of the document.

The `attach` option allows generated graphics to be embedded within the PDF using the `attachfile2` package; please load that package separately if you wish to use it. The `attach` option takes precedence over the `inline` option.

This package produces quite a number of output files, which by default are created in the same directory as the LaTeX document that is being compiled. To keep things more tidy, you can specify an output directory for these files by defining the `\asydir` command. For example, if you wish to store the figure files in the subdirectory `asytmp/`, the you would write `\renewcommand\asydir{asytmp}`.

Alternatively (and tentatively), you may write `dir=asytmp` in either the `asy` environment options or the options to `\asysetup`.

1

2.2 Commands for inserting Asymptote graphics

The main environment defined by the package is the `asy` environment, in which verbatim Asymptote code is placed that will be compiled for generating a graphic in the document. For example,

```
\begin{figure}
\begin{asy}[ <options> ]
<ASYMPTOTE CODE>
\end{asy}
\caption{...}\label{...}
```

If you have Asymptote code in a separate file, you can include it with the `\asyinclude[`⟨*options*⟩`]{`⟨*filename*⟩`}` command.

For Asymptote code that should be included in *every* graphic, define it using the `asydef` environment.

2.3 Graphics options

Both the `asy` environment and the `\asyinclude` command take optional parameters for controlling aspects of the graphics creation. In addition to locally setting `inline` and `attach`, the following options may also be used:

height Height of the figure

width Width of the figure

keepAspect Maintain aspect ratio [default true]

viewportheight Viewport height for 3D figures

viewportwidth Viewport width for 3D figures

These may also be set globally using the `\asysetup` command.

3 Processing the document

After running LaTeX on the document, it is necessary to process the Asymptote graphics so they can be included in the next compilation. The simplest procedure is a recipe such as

```
pdflatex mydoc
asy mydoc-*.asy
pdflatex mydoc
```

This technique will recompile each graphic every time, however. To only recompile graphics that have changed, use the `latexmk` tool. Asymptote is distributed with a `latexmkrc` configuration file; place this file in a place where `latexmk` will find it and your document may be compiled, including the `asy` compilations, with `latexmk mydoc` or `latexmk --pdf mydoc`.

4 Implementation

```
1 \def\Asymptote{{\tt Asymptote}}
```

```
2 \InputIfFileExists{\jobname.pre}{}{}
```

4.1 Allocations

Allocations

```
3 \newbox\ASYbox
4 \newcounter{asy}
```

```
5 \newwrite\AsyStream
6 \newwrite\AsyPreStream
```

```
7 \newif\ifASYinline
8 \newif\ifASYattach
9 \newif\ifASYkeepAspect
10 \ASYkeepAspecttrue
```

4.2 Packages

```
11 \RequirePackage{keyval}
12 \RequirePackage{ifthen}
13 \RequirePackage{color,graphicx}
```

Emulating packages   We cannot assume that Asymptote users have recent TEX distributions. (E.g., Fedora until recently still shipped teTeX.) So load ifpdf and ifxetex if they exist; otherwise, emulate them.

In due course, delete this code and just load the packages.

```
14 \IfFileExists{ifpdf.sty}{
15   \RequirePackage{ifpdf}
16 }{
17   \expandafter\newif\csname ifpdf\endcsname
18   \ifx\pdfoutput\@undefined\else
19     \ifcase\pdfoutput\else
20       \pdftrue
21     \fi
22   \fi
23 }
```

```
24 \IfFileExists{ifxetex.sty}{
25   \RequirePackage{ifxetex}
26 }{
27   \expandafter\newif\csname ifxetex\endcsname
28   \ifx\XeTeXversion\@undefined\else
29     \xetextrue
30   \fi
31 }
```

`\CatchFileDef`   Used for `\asyinclude`.

```
32 \IfFileExists{catchfile.sty}{
```

```
33   \RequirePackage{catchfile}
34 }{
35   \newcommand\CatchFileDef[3]{%
36     \begingroup
37     \everyeof{%
38       \ENDCATCHFILEMARKER
39       \noexpand
40     }%
41     \long\def\@tempa##1\ENDCATCHFILEMARKER{%
42       \endgroup
43       \def#1{##1}%
44     }%
45     #3%
46     \expandafter\@tempa\@@input #2\relax
47   }
48 }
```

Ensuring attachfile2 is loaded if [attach] is requested

```
49 \newif\if@asy@attachfile@loaded

50 \AtBeginDocument{%
51   \@ifpackageloaded{attachfile2}{\@asy@attachfile@loadedtrue}{}%
52   \let\asy@check@attachfile\asy@check@attachfile@loaded
53 }

54 \newcommand\asy@check@attachfile@loaded{%
55   \if@asy@attachfile@loaded\else
56     \PackageError{asymptote}{You must load the attachfile2 package}{^^J%
57       You have requested the [attach] option for some or all of your^^J%
58       Asymptote graphics, which requires the attachfile2 package.^^J%
59       Please load it in the document preamble.^^J%
60     }%
61   \fi
62 }

63 \newcommand\asy@check@attachfile{%
64   \AtBeginDocument{\asy@check@attachfile@loaded}%
65   \let\asy@check@attachfile\@empty
66 }
```

Macros

```
67 \def\csarg#1#2{\expandafter#1\csname#2\endcsname}
```

4.3   Package options

```
68 \DeclareOption{inline}{%
69   \ASYinlinetrue
70 }
71 \DeclareOption{attach}{%
72   \asy@check@attachfile
```

```
73    \ASYattachtrue
74 }
75 \ProcessOptions*

76 \def\asydir{}
77 \def\ASYprefix{}
```

4.4   Testing for PDF output

Note this is not quite the same as `\ifpdf`, since we still want PDF output when using XeTeX.

```
78 \newif\ifASYPDF
79 \ifxetex
80    \ASYPDFtrue
81 \else
82    \ifpdf
83       \ASYPDFtrue
84    \fi
85 \fi
86 \ifASYPDF
87    \def\AsyExtension{pdf}
88 \else
89    \def\AsyExtension{eps}
90 \fi
```

4.5   Bug squashing

```
91 \def\unquoteJobname#1"#2"#3\relax{%
92    \def\rawJobname{#1}%
93    \ifx\rawJobname\empty
94       \def\rawJobname{#2}%
95    \fi
96 }
97 \expandafter\unquoteJobname\jobname""\relax
```

Work around jobname bug in MiKTeX 2.5 and 2.6: Turn stars in file names (resulting from spaces, etc.) into minus signs

```
98 \def\fixstar#1*#2\relax{%
99    \def\argtwo{#2}%
100   \ifx\argtwo\empty
101      \gdef\Jobname{#1}%
102   \else
103      \fixstar#1-#2\relax
104   \fi
105 }
106 \expandafter\fixstar\rawJobname*\relax
```

Work around bug in dvips.def: allow spaces in file names.

```
107 \def\Ginclude@eps#1{%
108   \message{<#1>}%
109   \bgroup
110   \def\@tempa{!}%
```

```
111 \dimen@\Gin@req@width
112 \dimen@ii.1bp\relax
113 \divide\dimen@\dimen@ii
114 \@tempdima\Gin@req@height
115 \divide\@tempdima\dimen@ii
116   \special{PSfile=#1\space
117     llx=\Gin@llx\space
118     lly=\Gin@lly\space
119     urx=\Gin@urx\space
120     ury=\Gin@ury\space
121     \ifx\Gin@scalex\@tempa\else rwi=\number\dimen@\space\fi
122     \ifx\Gin@scaley\@tempa\else rhi=\number\@tempdima\space\fi
123     \ifGin@clip clip\fi}%
124   \egroup
125 }
```

4.6   Input/Output

```
126 \immediate\openout\AsyPreStream=\jobname.pre\relax
127 \AtEndDocument{\immediate\closeout\AsyPreStream}

128 \def\WriteAsyLine#1{%
129   \immediate\write\AsyStream{\detokenize{#1}}%
130 }

131 \def\globalASYdefs{}
132 \def\WriteGlobalAsyLine#1{%
133   \expandafter\g@addto@macro
134   \expandafter\globalASYdefs
135   \expandafter{\detokenize{#1^^J}}%
136 }
```

4.7   Commands for verbatim processing environments

```
137 \def\ProcessAsymptote#1{%
138   \begingroup
139   \def\CurrentAsymptote{#1}%
140   \let\do\@makeother \dospecials
141   \@makeother\^^L% and whatever other special cases
142   \catcode`\ =10
143   \endlinechar`\^^M \catcode`\^^M=12 \xAsymptote
144 }
```

Need lots of comment chars here because ⟨*line end*⟩ is no longer a space character.
```
145 \begingroup
146   \catcode`\^^M=12 \endlinechar=-1\relax%
147   \gdef\xAsymptote{%
148     \expandafter\ProcessAsymptoteLine%
149   }
150   \gdef\ProcessAsymptoteLine#1^^M{%
151     \def\@tempa{#1}%
152     {%
153       \escapechar=-1\relax%
```

```
154        \xdef\@tempb{\string\\end\string\{\CurrentAsymptote\string\}}%
155      }%
156      \ifx\@tempa\@tempb%
157        \edef\next{\endgroup\noexpand\end{\CurrentAsymptote}}%
158      \else%
159        \ThisAsymptote{#1}%
160        \let\next\ProcessAsymptoteLine%
161      \fi%
162      \next%
163    }
164  \endgroup
```

4.8   User interface

```
165  \newcommand\asy[1][]{%
166    \stepcounter{asy}%
167    \setkeys{ASYkeys}{#1}%
```

Disable the "inline" option if "attach" is enabled:

```
168    \ifASYattach
169      \ASYinlinefalse
170    \fi
171    \ifx\asydir\empty\else
172      \def\ASYprefix{\asydir/}%
173    \fi
174    \immediate\write\AsyPreStream{%
175      \noexpand\InputIfFileExists{%
176        \ASYprefix\noexpand\jobname-\the\c@asy.pre}{}{}%
177    }
178    \asy@write@graphic@header
179    \let\ThisAsymptote\WriteAsyLine
180    \ProcessAsymptote{asy}%
181  }

182  \def\endasy{%
183    \asy@finalise@stream
184    \asy@input@graphic
185  }

186  \def\asy@write@graphic@header{%
187    \immediate\openout\AsyStream=\ASYprefix\jobname-\the\c@asy.asy\relax
188    \gdef\AsyFile{\ASYprefix\Jobname-\the\c@asy}%
189    \immediate\write\AsyStream{%
190      if(!settings.multipleView) settings.batchView=false;^^J%
191      \ifxetex
192        settings.tex="xelatex";^^J%
193      \else\ifASYPDF
194          settings.tex="pdflatex";^^J%
195      \fi\fi
196      \ifASYinline
197        settings.inlinetex=true;^^J%
198        deletepreamble();^^J%
```

```
199     \fi
200     defaultfilename="\Jobname-\the\c@asy";^^J%
201     if(settings.render < 0) settings.render=4;^^J%
202     settings.outformat="";^^J%
203     \ifASYattach
204       settings.inlineimage=false;^^J%
205       settings.embed=false;^^J%
206       settings.toolbar=true;^^J%
207     \else
208       settings.inlineimage=true;^^J%
209       settings.embed=true;^^J%
210       settings.toolbar=false;^^J%
211       viewportmargin=(2,2);^^J%
212     \fi
213     \globalASYdefs
214   }%
215 }
216 \def\asy@expand@keepAspect{%
217   \ifASYkeepAspect keepAspect=true%
218   \else keepAspect=false%
219   \fi%
220 }

221 \def\asy@finalise@stream{%
```

Setting `size()`. Only inserted if one of the dimensions is set explicitly (i.e., if both height and width are not empty).

```
222   \ifx\ASYwidth\@empty
223     \ifx\ASYheight\@empty
224       % write nothing!
225     \else
226       \immediate\write\AsyStream{size(0,\ASYheight,\asy@expand@keepAspect);}%
227     \fi
228   \else
229     \ifx\ASYheight\@empty
230       \immediate\write\AsyStream{size(\ASYwidth,0,\asy@expand@keepAspect);}%
231     \else
232       \immediate\write\AsyStream{size(\ASYwidth,\ASYheight,\asy@expand@keepAspect);}%
233     \fi
234   \fi
```

Setting `viewportsize=()`. Same logic as for `size()`. Note that `\ASYviewportwidth` varies according to `\ASYattach`, by default, so it needs to be fully expanded before checking if it equals zero.

```
235   \edef\ASYviewportwidth{\ASYviewportwidth}% locally "freeze" viewportwidth
236   \ifx\ASYviewportwidth\@empty
237     \ifx\ASYviewportheight\@empty
238       % write nothing!
239     \else
240       \immediate\write\AsyStream{viewportsize=(0,\ASYviewportheight);}%
241     \fi
```

```
242    \else
243      \ifx\ASYviewportheight\@empty
244        \immediate\write\AsyStream{viewportsize=(\ASYviewportwidth,0);}%
245      \else
246        \immediate\write\AsyStream{%
247          viewportsize=(\ASYviewportwidth,\ASYviewportheight);}%
248      \fi
249    \fi
250    \immediate\closeout\AsyStream
251 }
252 \def\asy@input@graphic{%
253    \ifASYinline
254      \IfFileExists{"\AsyFile.tex"}{%
255        \catcode`:=12\relax
256        \@@input"\AsyFile.tex"\relax
257      }{%
258        \PackageWarning{asymptote}{file `\AsyFile.tex' not found}%
259      }%
260    \else
261      \IfFileExists{"\AsyFile.\AsyExtension"}{%
262        \ifASYattach
263          \ifASYPDF
264            \IfFileExists{"\AsyFile+0.pdf"}{%
265              \setbox\ASYbox=\hbox{\includegraphics[hiresbb]{"\AsyFile+0".pdf}}%
266            }{%
267              \setbox\ASYbox=\hbox{\includegraphics[hiresbb]{"\AsyFile".pdf}}%
268            }%
269          \else
270            \setbox\ASYbox=\hbox{\includegraphics[hiresbb]{"\AsyFile.eps"}}%
271          \fi
272          \textattachfile{\AsyFile.\AsyExtension}{\phantom{\copy\ASYbox}}%
273          \vskip-\ht\ASYbox
274          \indent
275          \box\ASYbox
276        \else
277          \ifASYPDF
278            \includegraphics[hiresbb]{"\AsyFile".pdf}%
279          \else
280            \includegraphics[hiresbb]{"\AsyFile.eps"}%
281          \fi
282        \fi
283      }{%
```

3D PRC figures require inline mode.

```
284        \IfFileExists{"\AsyFile.tex"}{%
285          \catcode`:=12
286          \@@input"\AsyFile.tex"\relax
287        }{%
288          \PackageWarning{asymptote}{%
289            file `\AsyFile.\AsyExtension' not found%
```

```
290          }%
291        }%
292      }%
293    \fi
294 }
295 \def\asydef{%
296    \let\ThisAsymptote\WriteGlobalAsyLine
297    \ProcessAsymptote{asydef}%
298 }
299 \newcommand\asyinclude[2][]{%
300    \begingroup
301    \stepcounter{asy}%
302    \setkeys{ASYkeys}{#1}%
303    \ifASYattach
304      \ASYinlinefalse
305    \fi
306    \ifx\asydir\empty\else
307      \def\ASYprefix{\asydir/}%
308    \fi
309    \immediate\write\AsyPreStream{%
310      \noexpand\InputIfFileExists{%
311        \ASYprefix\noexpand\jobname-\the\c@asy.pre}{}{}%
312    }%
313    \asy@write@graphic@header
314    \IfFileExists{#2.asy}{%
315      \CatchFileDef\@tempa{#2.asy}{%
316        \let\do\@makeother
317        \dospecials
318        \endlinechar=10\relax
319      }%
320    }{%
321      \IfFileExists{#2}{%
322        \CatchFileDef\@tempa{#2}{%
323          \let\do\@makeother
324          \dospecials
325          \endlinechar=10\relax
326        }%
327      }{%
328        \PackageWarning{asymptote}{file #2 not found}%
329      }%
330    }%
331    \immediate\write\AsyStream{\unexpanded\expandafter{\@tempa}}%
332    \asy@finalise@stream
333    \asy@input@graphic
334    \endgroup
335 }
336 \newcommand{\ASYanimategraphics}[5][]{%
337    \IfFileExists{_#3.pdf}{%
338      \animategraphics[{#1}]{#2}{_#3}{#4}{#5}%
```

```
339   }{}%
340 }
```

4.9   Keys for graphics processing

```
341 \newcommand\asysetup[1]{\setkeys{ASYkeys}{#1}}

342 \define@key{ASYkeys}{dir}{%
343   \def\asydir{#1}%
344 }
345 \def\ASYwidth{}
346 \define@key{ASYkeys}{width}{%
347   \edef\ASYwidth{\the\dimexpr#1\relax}%
348 }
349 \def\ASYheight{}
350 \define@key{ASYkeys}{height}{%
351   \edef\ASYheight{\the\dimexpr#1\relax}%
352 }
353 \define@key{ASYkeys}{keepAspect}[true]{%
354   \ifthenelse{\equal{#1}{true}}
355     {\ASYkeepAspecttrue}
356     {\ASYkeepAspectfalse}%
357 }
358 \def\ASYviewportwidth{\ASYdefaultviewportwidth}
359 \def\ASYdefaultviewportwidth{%
360   \ifASYattach \@empty \else \the\linewidth \fi
361 }
362 \define@key{ASYkeys}{viewportwidth}{%
363   \edef\ASYviewportwidth{\the\dimexpr#1\relax}%
364 }
365 \def\ASYviewportheight{}
366 \define@key{ASYkeys}{viewportheight}{%
367   \edef\ASYviewportheight{\the\dimexpr#1\relax}%
368 }

369 \define@key{ASYkeys}{inline}[true]{%
370   \ifthenelse{\equal{#1}{true}}
371     {\ASYinlinetrue}
372     {\ASYinlinefalse}%
373 }
374 \define@key{ASYkeys}{attach}[true]{%
375   \ifthenelse{\equal{#1}{true}}
376     {\ASYattachtrue}
377     {\ASYattachfalse}%
378 }
```